Experimental large-scale review of attractors for detection of potentially unwanted applications

Accepted Manuscript

Title: Experimental large-scale review of attractors for detection of potentially unwanted applications

Author: Vlasta Stavova, Lenka Dedkova, Vashek Matyas, Mike Just, David Smahel, Martin Ukrop

PII: S0167-4048(18)30164-0
DOI: https://doi.org/10.1016/j.cose.2018.02.017
Reference: COSE 1302

To appear in: *Computers & Security*

Received date: 22-12-2017
Revised date: 21-2-2018
Accepted date: 25-2-2018

Please cite this article as: Vlasta Stavova, Lenka Dedkova, Vashek Matyas, Mike Just, David Smahel, Martin Ukrop, Experimental large-scale review of attractors for detection of potentially unwanted applications, *Computers & Security* (2018), https://doi.org/10.1016/j.cose.2018.02.017.

Experimental large-scale review of attractors for detection of potentially unwanted applications

Vlasta Stavova[a], Lenka Dedkova[b], Vashek Matyas[a], Mike Just[c], David Smahel[a], Martin Ukrop[a]

[a]*Faculty of Informatics, Masaryk University, Czech Republic*

[b]*Faculty of Social Studies, Masaryk University, Czech Republic*

[c]*School of Mathematical & Computer Sciences, Heriot-Watt University, United Kingdom*

Vlasta Stavova (vlasta.stavova@mail.muni.cz) is a PhD candidate in the Centre for Research on Cryptography and Security in the Faculty of Informatics at Masaryk University, Brno, Czech Republic.

Lenka Dedkova (ldedkova@fss.muni.cz) is a postdoc researcher at Institute for Research on Children, Youth and Family at Faculty of Social Sciences, Masaryk University, Brno, Czech Republic.

Vashek Matyas (matyas@fi.muni.cz) is a Professor in the Centre for Research on Cryptography and Security at the Faculty of Informatics at Masaryk University, Brno, Czech Republic.

Mike Just (m.just@hw.ac.uk) is an Associate Professor and Deputy Head of Computer Science, School of Mathematical and Computer, Heriot-Watt University, Scotland.

David Smahel (davsmahel@gmail.com) is a Professor in the Institute for Research of Children, Youth and Family at the Faculty of Social Studies at Masaryk University, Brno, Czech Republic.

Martin Ukrop (mukrop@mail.muni.cz) is a PhD candidate in the Centre for Research on Cryptography and Security at the Faculty of Informatics at Masaryk University, Brno, Czech Republic.

Abstract

While malicious software (malware) is designed to disrupt or damage computer systems, potentially unwanted applications (PUAs) combine useful features with less desirable ones, such as adware or spyware. Unlike anti-malware solutions, removing PUAs can be controversial, for both the PUA owners and also the users who might wish to accept the PUA features. Thus,

Email addresses: vlasta.stavova@mail.muni.cz (Vlasta Stavova), ldedkova@fss.muni.cz (Lenka Dedkova), matyas@fi.muni.cz (Vashek Matyas), m.just@hw.ac.uk (Mike Just), davs@mail.muni.cz (David Smahel), mukrop@mail.muni.cz (Martin Ukrop)

solutions for removing PUAs require users to make their removal decisions. In this paper we investigate the effectiveness of 15 screen variants that use different "security warning attractors" designed to encourage users to enable PUA detection when they are installing a security software solution from the online security software company ESET. Our live field study with close to 750,000 software installations by end users in 222 countries shows that a small change of switching the order of the options presented using radio buttons and offering the "enable detection" option first was the most effective (and was later set as the option of choice by ESET). The chosen approach led to a significant reduction of non-consenting users from 17.9% to 11.1%. Other features, such as the use of colours and pictorials, which have previously demonstrated their effectiveness with more traditional SSL security warnings, did not yield significant improvements for enabling PUA detection.

Keywords: usable security; potentially unwanted application; attractor; security software; user decision

1. Introduction

Potentially unwanted applications (PUAs, a.k.a. potentially unwanted programs, PUPs), cover several arguably malicious families of software such as adware, spyware, pornware, bundleware or junkware. Differing from malicious software (*malware*), PUAs often combine a potentially useful feature with arguably less desirable features that deliver unwanted ads, monitor users' behaviour or collect their data [1].

Many online security software solutions (e.g., endpoint antivirus with some additional features) include a service to detect and alert users about PUAs targeting their devices. However, automatically classifying an application as a PUA can be challenging, in part due to the different perceptions of what constitutes an "unwanted" application. For example, some PUAs might be knowingly installed by users, such as with browser toolbars that are sometimes packaged with software. Even benign applications (such as remote desktop controllers or various registry cleaners) can contain functions that would be identified as unwanted by some users. However, automatically classifying such software as "unwanted" for all users can create confusion that could cause users to lose trust in PUA classification decisions. Further, as several example encounters have shown, well-established adware companies do not hesitate to sue security software vendors for automatically classifying their software as adware [2].

These circumstances pose a unique challenge for security software vendors who want to protect their users and facilitate PUA detection, but who might be legally restricted from automatically removing a PUA. The approach chosen by vendors has been to involve users in the decision of labeling applications as unwanted, before removing them from their system. In this way, the approach to involve the user in PUA decisions is similar to what is done for security warnings, for example for phishing [3], SSL warnings [4] or malware [5]. While there has been much recent focus on such security warnings, there has surprisingly been little research undertaken so far in user decisions for PUA warnings.

There are two stages of user involvement with PUAs. First, there is a choice about whether or not to enable the detection of PUAs in users' security software. Second, if PUA detection is enabled, there are decisions about whether or not to accept or reject an individual application identified as a PUA. In this paper, we focus on the first stage.

A 2016 survey [6] with 2,022 participants found that 73% of people who changed default security software settings (from 41.2% of all research participants) also enabled PUA detection. However, the study used users' self-reports, which may lead to inaccurate estimation of PUA detection enablement.

In our previous study [1] we collected system installation data from a large set of beta testers and found out that overall, 74.7% of beta testers enabled PUA detection. Since beta testers may differ from standard end users in terms of their IT abilities, and consequently their ICT related behaviour, repeating research with standard end users is necessary [7].

In this paper, we present the results of our study of the effectiveness of 15 variants with different "attractors" that were designed for encouraging users to enable the detection of PUAs. The attractors consist of different interface modifications, similar to those studied for security warnings.

Each of 748,795 end users were presented with a single randomly chosen variant when installing an online security software solution from security software company ESET. We report the decisions of our end-user participants to enable the detection of PUAs, as well as the time they spent on each screen to make their decision.

The following section describes the dataset, introduces our data cleaning and analytical strategy, and presents the 15 designed variants. Section 3 reports on the effect of the attractors on users' decisions to enable the detection of PUAs. In Section 4, the issue of time spent on the

variants' screens is examined. Study limitations and related research on PUAs are discussed in Section 5 and 6. Section 7 then concludes our article.

2. Methods

Our study was conducted in cooperation with ESET, an online security software company with over 100 million users in more than 200 countries and territories[1]. During the installation process, ESET presents a screen dialogue that asks users to either enable or disable the detection of PUAs. This step cannot be skipped and thus each user has to choose one option or the other, although it is possible to change this decision later in the software settings.

For our experiment, we prepared 15 different screen variants (14 new approaches and 1 control variant (A1) – see Figures 1 and 2) of the PUA enable/disable screen with various attractors. One variant was randomly selected for display to each user during the installation process. We selected and tested five basic approaches to guide the design of our 15 variants based on well-known user dialog and security warning design principles (see Subsection 6.2) The particular options and settings underwent both discussions between the research team and ESET, as well as formal approval processes of ESET.

Table 1 shows each approach (attractor type) along with the corresponding variants used with that approach. Some screens combined multiple features; see Table 2 for a concise description of what features were included in each variant. See Figures 1 and 2 for the visual representation of the PUA dialog variants.

The screens were written in English and were thus presented to the users during installation of the English version of the security software. Note that ESET did not impose any limitation on downloading the English software version to particular continents or countries. For each installation, we collected information about users' decisions to enable or disable PUA detection in the installation process, and the time the users spent on the screen with the PUA dialog. The data was collected from October 2016 to February 2017 and came from 748,795 end-user installations[2] of ESET security software solution for Microsoft Windows OS spread across 222 countries.

2.1. Analytical strategy

[1] https://www.eset.com/int/about/

[2] We refer to "end user installations" since we were not able to completely control multiple installations by the same user – see Subsection 2.2.

During the analysis, we first used the omnibus χ^2 test to discover whether there were overall differences in the rate of enabling PUA detection among the screen variants, and then planned to proceed with pairwise comparisons to examine the specific effects of each attractor. Analysing such large sample sizes leads to inflated significance tests, thus we also calculated Cramer's V (φ_c) to better assess the effect sizes; the value of 0.1 is considered as small, 0.3 as medium and 0.5 as a large effect size [8]. Next, we focused on the time spent on each screen. It is well known that users tend not to read the text in dialog windows [9], hence we were interested in examining whether users in our sample spent a sufficient amount of time on the screens in order to have the chance of actually reading them. Admittedly, merely spending enough time cannot be equated to reading the text, which poses a limit to the interpretation of the findings. However, spending significantly less than a "sufficient amount of time" can be interpreted as not reading the (entire) text. Thus, since the screens differed in the number of words (ranging from 29 to 123 words), we compared the time spent on each screen to the time that an average adult would spend reading the respective number of words. According to Taylor [10], the average reading speed for English is 200-250 words per minute. In order not to underestimate the users' reading speed, we chose to use the higher speed (250 words per minute) to calculate the needed time for each screen (see Table 3 and more details in Section 4). Then, a one-sample *t*-test was used to test the differences between the time needed (used as population average) and time actually spent on each screen.

2.2. Data cleaning

The original dataset obtained from ESET included data from 799,450 end-user installations (cases). To clean the dataset, we first excluded the cases with ESET's internal IP address (N = 275). Furthermore, in an attempt to remove multiple installations from the same computing device, the duplicate entries were deleted. These were identified by combining hardware features, IP addresses, and hashed MAC addresses (N = 50,380), leaving the final dataset of N = 748,795 installations (thereby removing approximately 6% of the entries from the original dataset).

In the original dataset, the "time spent" data from each screen was highly skewed: the values ranged from 0 seconds to 563 hours with a mean of 48.41 seconds (SD = 3,378.549), median of 10 seconds and mode of 3. The longer durations may have had two causes: users could have left the dialog window open and the computer running while they were doing something

completely different, or the time could have been wrongly recorded in some cases. This was however true for only a small amount of data. To clean the time data of these cases, we thus omitted the highest 1% of the values from each screen, removing 7,529 cases from the time analyses. For the resultant data, the average "time spent" duration decreased to 16.625 seconds (*SD* = 21.976, range 0 to 299 seconds). The mode and median remained the same.

3. Findings

The omnibus χ^2 test showed significant differences among the tested variants ($\chi^2(14)$ = 2,984.057; $p < 0.001$), hence we continued with the planned pairwise tests. We performed two levels of comparison. Firstly, we compared screens that were similar in appearance, with the exception of presence or absence of a distinct attractor (for example, presence or absence of a pictorial), which we discuss in each of the subsections below. Secondly, we compared all variants of the attractors with the control, for which results are presented in Table 2. This table also shows the detection rates across examined screen variants. Each subsection below describes detailed results for each of our five design approaches.

3.1. Text structure

Paragraph text. The following paragraph was used to explain PUAs to end users: "*ESET can detect potentially unwanted applications and ask for confirmation before they install. Potentially unwanted applications might not pose a security risk but they can affect computer's performance, speed and reliability, or cause changes in behaviour. They usually require user's consent before installation.*" We tested whether the presence (screen B2) or absence (E4) of this explanation (when used with the same order of options presented, using radio buttons for enabling or disabling PUA detection) made a difference and found a negligible effect ($\chi^2(1) = 19.887$; $\varphi = 0.014$, $p < 0.001$), slightly in favor of presence of the explanation (82.2% vs. 81.1%).

Bullet points. We re-arranged the paragraph text explaining PUAs into bullets to increase its readability (the content remained the same). The bullet points in the text version (screen A2) had a negligible positive effect on the detection rate that was 0.5% higher ($\chi^2(1) = 4.515$; $\varphi = 0.007$, $p < 0.05$) over the use of a paragraph (screen A1).

3.2. Purpose stressing

PUA example. The example of a potentially unwanted application risk was presented on screen B1. The exact wording was *"**For example**, they may change your web browser's web page and*

search settings." When compared to the control screen (A1), no significant difference was found ($p > 0.05$).

External link. Two versions of the external link were used: one read *"What is a potentially unwanted application?"* (B2) and we also explored the effect of less text in the link description *"Why do we ask?"* (B3). Both hyperlinks lead to the same page[3] with a detailed explanation of PUAs and examples provided, and also what happens when a PUA is detected. First, we compared screens B2 and B3 with the control variant (A1). Next, we compared screens B2 and B3 to see whether the formulation of the link made a difference. These comparisons did not reveal any significant differences ($p > 0.05$). This suggests that neither the presence nor wording of the external link impacts a user's decision to enable PUA detection.

Further, we explored the effect of clicks on the link. In total, six screens included a hyperlink (N = 299,419) with 7.5% (N = 22,522) of users clicking on it. Screens E3 and E4, that did not include the PUA explanation, resulted in 10% and 11% of users clicking on the link, whereas for other screens, the average click rate was approximately 6% (the differences between screens E3 and E4, as well as the differences between screens B2, B3, E1 and E2 were not significant, while the differences between E3 and E4 and the other screens (including screens without hyperlink) were significant with φ ranging from 0.089 to 0.096). In all screens with links, clicking users enabled PUA detection more often, with very similar effect sizes on each screen – hence, for brevity, we present only overall rates across all six screens: 88.7% of users clicking on the link enabled PUA detection, whereas 81.0% of non-clicking users did ($\chi^2(1) = 826.897$, $\varphi = 0.053$, $p < 0.001$).

This might lead one to conclude that providing more information through a hyperlink is an effective way for users to learn about PUAs, while also increasing the PUA enablement rate. However, note that it was only a small percentage of participants (6%) who clicked on the explanation link. Therefore, even though there was a higher tendency of these users to choose to enable PUA detection, we don't believe that there is currently enough data to confirm this as a viable design strategy.

3.3. Attention raising

Graphic attractors were used in three screens: C1 (red signal word "Notice" at the start of the text description), C2 (triangle with a black exclamation mark – the ANSI-inspired pictorial),

[3]http://support.eset.com/kb2629/?locale=en_US

and C3 (octagon-framed exclamation mark – the company's warning pictorial). When compared to the control screen (A1), no significant differences were found ($p > 0.05$). Moreover, no differences were found between the two pictorials: users did not act differently on screens using the company warning sign and the standard ANSI-inspired pictorial, either in terms of enabling PUA detection, or in terms of the time spent on the screen. The ANSI-inspired pictorial is based on the sign that is widely used as a safety sign in both industry and software development and therefore it is connected with a perception of risk. For example, a yellow triangle with an exclamation mark serves as a warning sign for a low battery in the Android OS.

In our experiment, none of the graphic attractors influenced user choice – neither in terms of the PUA detection enabling rate nor in the time spent on the screen. Their role as attractors is problematic, especially in cases where the graphic covers only a small piece of the screen.

3.4. Options order

Query wording and order of presented options. We compared the control screen (A1; *disable/enable*) with the variant in which the options were presented in reverse order (*enable/disable*; screen D1) and found a significant difference ($\chi^2(1) = 939.725$, $p < 0.001$). Screen D1 is also the screen that prompted the most users to enable PUA detection across all screen variants (88.94%; see Table 2). Although the effect size is rather small ($\varphi = 0.097$, $p < 0.001$), increasing the number of users who enable PUA detection by 7 percentage points is a substantial improvement (representing more than 53,000 of our users).

We also evaluated the use of alternate verbs, *detect/don't detect* (and vice versa). While the screen with *don't detect* presented as the first option (D2) lowered the users' enablement of PUA detection ($\chi^2(1) = 133.621$; $\varphi = -.037$, $p < 0.001$) by nearly 3 percentage points, presenting the preferred, positive, option of *detect* first (D3), increased the enablement of PUA detection by almost 5 percentage points ($\chi^2(1) = 457.175$; $\varphi = -0.068$, $p < 0.001$). However, the verb *enable*, presented in the first position, performed significantly better than the verb *detect* ($\chi^2(1) = 88.281$; $\varphi = 0.030$, $p < 0.001$). Based on these findings, ESET has since started to use screen D1 in their installation process.

3.5. Presence of another user dialog

This included screens E3 vs. E4 ($\chi^2(1) = 74.394$, $\varphi = 0.027$, $p < 0.001$) and screens E1 vs. E2 ($\chi^2(1) = 3.861$, $p < 0.05$, $\varphi = 0.006$, $p < 0.005$). In both comparisons, the screen without the

other dialog caused slightly more users to enable PUA detection. This result suggests that it may be beneficial not to present multiple queries to users within a single installation screen.

3.6. Combinations

Since we initially expected that some combinations might strengthen the effect of the attractors, we designed two combined variants (screens E1 and E2). They combine the structure and purpose approach by including bulleted text and a hyperlink. In addition, one variant did not contain an additional user dialog (E1) whereas the other did (E2). Somewhat surprisingly, both of these variants had the opposite effect when compared to the control screen – both slightly decreased the rate of enabling PUA detection (screen E1: $\chi^2(1) = 40.499$; $\varphi = -0.020$, $p < 0.001$; screen E2: $\chi^2(1) = 69.634$; $\varphi = -0.026$, $p < 0.001$).

Overall, our findings suggest that the text itself and visual aids in the form of warning symbols do not impact users' choice to enable PUA detection in our study; only presenting the preferred option in the first position mattered substantially. This raises the question of whether the users actually read all the text on the screen. Therefore, we further examined the users' behaviour with a specific focus on the time spent on each screen.

4. Time spent on screen

Before analysing the differences among users and the variants for the time spent on each screen, we first omitted those users who clicked on the hyperlink presented (for those variants that included a hyperlink), since these users were directed to a secondary webpage with a PUA explanation which thus prolonged their recorded time on the screen ($N_{clicked} = 22{,}522$).

4.1. Differences across the screen variants

We observed differences in the time spent on the screen per each variant ($F(14) = 365.114$, $p < 0.001$; see Table 3). Interestingly, the screens that users on average spent most time on were the screens with the smallest number of words – which is contrary to the expected result if we were to believe that the users actually read the text (i.e., the greater the number of words, the more time a user should spend on the screen). To better assess the differences among the screens and users' reading habits, we compared the average time users spent on each screen to the time that would be needed to read the screen according to the literature with a one-sample *t*-test (see Subsection 2.1 for more details). All examined differences were significant ($p < 0.001$), and for most screens, the resultant difference was negative. This indicates that users spent substantially less time on a screen than would be needed to read the text – these differences

ranged from −2.48 (in screen E4) to −13.83 seconds (screen B1). Users spent more time than necessary to read the text on two screens only – 10.28 seconds more on screen E3 and 5.88 seconds more on screen E1). For a better presentation of the time data, we coded a binary variable to denote whether the user had or did not have enough time to read the content on the screen (see also Table 3). For screen E3, consisting of 29 words, nearly 76% of users had enough time. From that, the rate sharply drops to less than half (46.3%) of users on screen E1, followed by 25.3% on screen E4 and decreasing down to 12% on screen D1.

Regarding the number of words on the most read screens – the most "successful" screens, E1 and E3, are the only screens that did not include another user dialog, and thus consisted of the least amount of text (64 and 29 words, respectively). It seems that users are willing to devote their energy to reading when they feel the costs are not too high (in terms of their effort and time). Interestingly, the third most read screen had 21% fewer readers than the second. This is a large decrease considering that the screen is only 11 words longer. It does, however, include another user dialog. We thus hypothesize that not only the text amount but also a number of different user dialogs make the difference – i.e., the users seem to spend time reading the text on potentially unwanted application detection when the text is short and they are not distracted (or discouraged) by the presence of other user dialog.

4.2. Differences between users who enabled and did not enable PUA detection

We further examined whether there are differences in time spent on the screen between those users who enabled PUA detection and those who did not. We calculated a *t*-test for independent samples for each screen. Surprisingly, the differences were negligible: despite sometimes statistically significant at *p*-value 0.001, they did not exceed 2.2 seconds for any screen variant, and the average differences between those enabling and not enabling PUA detection across all screens were only 1.31 seconds. Another surprising finding related to enablement was that it was those users who decided not to enable PUA detection who seemed to spend a bit more time on the screens. This tendency can be observed also in the numbers of users who spent a sufficient amount of time on the screen – the two lowest rates (12% and 13.5%) were obtained on the two screens with the highest number of users enabling PUA detection (screen D1 and D3).

This finding suggests that users do not base their decisions to enable PUA detection on fully reading and understanding the text presented on the screen, but rather the opt for the

quickest way forward with the installation process. Under such conditions, designing the dialog screen in a way that prompts most users to choose the preferred setting (even without reading the text) bears crucial importance. We would thus advise security software companies to pay particular attention to design features and warn them not to underestimate design effects.

5. Study limitations

Our study has several limitations. First, we focused on the form rather than content of the text, being limited by the cooperating company, ESET, namely their legal and marketing staff. The text comprehensibility represents an important factor in users' decisions and we would urge researchers to design and test various text alterations that would increase users' understanding while still remaining general enough to prevent potential legal issues. Second, we used only the English version of the online antivirus software and proposed screen variants. This may play a role in comprehension in countries where English is not a mother tongue. Third, the limitation set by ESET was to do only small, minor changes in the PUA user dialog. Thus, we were not able to influence the overall workflow of program installation.

6. Related work

Related work includes PUA-oriented research and also research on influencing user decisions towards the safe choice through attractors, i.e., interface modifications that attempt to attract users' attention and change their behaviour.

6.1. PUA scene

PUAs are significant issues. For example, Kwon et al. [11] analyzed a non-trivial subset of VirusTotal reports from 2013, and they identified that PUAs are up to 10 times more common than malware.

PUAs also dominate in the category of signed potentially malicious applications. Kotzias et al. [12] analyzed 356,931 software samples from 2006 to 2015, collected mainly from VirusShare and other publicly available data sources. The most signed samples are PUAs (88%-95%) whereas malware is nearly never signed. They also observed that the number of PUAs in their dataset is steadily growing over time since 2011. PUAs are also highly prevalent in Pay-Per-Install services. Kotzias et al. [13] measured PUA prevalence on real hosts finding that 54% have PUAs installed. Their dataset contained more than 8 billion events on 3.9 million real hosts in a period of 19 months, capturing only signed PUA executables. There is also a high risk of PUA consequent spread because 65% PUA downloads are performed by another PUA.

6.2. Motivation for designed screens

Research on the effectiveness of security warnings has studied how different interface features, such as text, pictorials, and symbols can used to increase users' attention and adherence to security warnings, as well as their comprehension. For example, previous work [14, 5] on SSL warnings demonstrates the effectiveness of well-designed warnings. In this article, we've investigated whether some of these features can help to encourage users to enable PUA detection.

For example, research suggests that text structure impacts its comprehension, since warning text in bullets or in an outline form is considered more readable than continuous text [15]. More recently, Bravo-Lillo et al. [16] found that a detailed explanation serves as a "bad attention" attractor. However, others [17] note that a warning containing a "purpose string" has a higher impact on a user over the warning without any purpose. Surprisingly, an effect of different content in a purpose string is statistically insignificant. In our study, providing a purpose and an explanation led to a significant increase in the PUA enablement rate (in comparison with screens where no explanation was provided).

While pictorials were originally important in physical security, they are currently also widely used in user dialogs, especially in the context of web browsers. Bold print, high contrast and pictorial symbols enhance the salience of visual warnings [18]. In addition, colour in a warning can increase its ability to attract attention [19]. Concerning our study, visual salience did not lead to an increase in the desired behaviour of enabling PUA detection.

Felt et al. [20] discuss icon shape in communicating the level of risk for security warnings in browsers. They found that an exclamation mark in both a triangle and a circle was perceived as not connected with security [20]. However, using a triangle and a circle together with an exclamation mark had the best results. As for the triangle, all colours seem reasonably well distributed, the majority in connection with orange, red and blue colour. Similar effects were not observed with the use of pictorials and icons in our study. In particular, we did not observe a statistically significant difference in variants with and without a colourful pictorial. We also did not observe differences in the PUA enablement rate for screens containing either of the two pictorials that we investigated. The triangle symbol that we used was inspired by the ISO graphical symbol as a typical example of a safety alert symbol that many people might be

familiar with. Originally, it was a symbol that indicated a potential personal injury hazard [21]. The ESET company's own warning pictorial is widely used in the company's security software.

Research [22] has also recommended the use of a signal word in warnings to increase their effectiveness. Warnings printed in red (compared to black) led to improved noticeability [23]. On the other hand, the usage of colour only is problematic since 8% men are colourblind [24]. The signal word "Notice" is recommended for use by the American National Standards Institute Z535 Standards on Safety Signs and Colours [25]. In our study, the word "Notice" in a red colour did not lead to significant increase in the PUA enablement rate.

With respect to the time spent on various screens, Akhawe et al. [5] compared the amount of time spent on a security attractor by users to their reaction to the attractor, e.g., to adhere to the attractor. They found that users who click through (ignore) browser security warnings spend less time on the warnings. While our results were not significant, we found that users who chose not to enable PUA detection spent a bit more time on the attractor.

7. Conclusions

We analyzed several attractors in 15 screen variants of a user dialog for enabling PUA detection. The variants cover both textual and visual aspects previously tested and approved in warning design, such as attention raising, text structuring, purpose stressing, etc. We cooperated with the security software company ESET and collected data from about 750,000 end-user installations with our variants displayed during the security software installation.

Variant D1 scored the greatest improvement in terms of the number of users who enabled PUA detection, when compared to an original control dialog. This new variant is currently being used by online security software company ESET in their security products. We encourage further research in the selection of options for user interface settings as we show that even minor changes can cause a significant impact.

Summarizing the performance of features that we tested in our screen variants, the order of options for deciding to enable or disable PUA detection matters the most. In our study, presenting the enable option first (as with D1 above) leads to the highest number of users enabling PUA detection. In general, "attention raising" features do not lead to a significant improvement nor did differences in text structuring. Similarly, combinations of features that we tested did not lead to any significant increase of users enabling PUA detection.

We also found a small portion of users to be interested in additional information regarding the user dialog (e.g., by following a hyperlink). These users then tend to enable PUA detection more than the others.

Users are negatively influenced by other user dialogs that are present or absent on/from the screen, even when the other dialog response is pre-checked by default and does not require any further user interaction. We believe that presenting a short single user dialog on the screen has a higher chance of being read by users. This would be an appropriate recommendation when the overall number of screens is low (in our case, the installation process consisted of five screens that required some user attention), but might be perceived negatively in a longer screen sequence.

As an interesting observation, we were not able to determine whether the time spent on the screen generally leads to increasing the PUA enablement rate. In fact, we observed a slight decrease in this rate with more time spent on screen. This may be caused by necessarily vague formulations used to describe PUAs – the descriptions could not include very specific effects of PUAs and not even any concrete examples, in order to avoid potential legal battles. We encourage future research to examine users' understandings of such descriptions in more depth, such as in strict experimental settings.

Acknowledgement

We thank Masaryk University (project MUNI/E/1281/2016) and ESET.

References

[1] V. Stavova, V. Matyas, M. Just, On the impact of warning interfaces for enabling the detection of Potentially Unwanted Applications, in: Euro Usable Security (EuroUSEC) Workshop Programme, 2016, ISBN: 1-891562-45-2.

[2] M. Masnick, Gator Threatening Those Who Call Their Application Spyware, https://www.techdirt.com/articles/20031022/1420248_F.shtml, accessed: 2018-02-18 (1998).

[3] S. Egelman, S. Schechter, The importance of being earnest [in security warnings], in: International Conference on Financial Cryptography and Data Security, Springer, 2013, pp. 52–59.

[4] S. Fahl, M. Harbach, T. Muders, M. Smith, L. Baumgärtner, B. Freisleben, Why eve and mallory love android: an analysis of android ssl (in)security, in: ACM Conference on Computer and Communications Security, 2012.

[5] D. Akhawe, A. P. Felt, Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness., in: Proceedings of the USENIX Security Symposium, Vol. 13, 2013.

[6] AV Comparatives, IT Security Survey 2016, http://www.av-comparatives.org/wp-content/uploads/2016/02/security_survey2016_en.pdf, accessed: 2018-02-18.

[7] V. Stavova, L. Dedkova, M. Ukrop, V. Matyas, A large-scale comparative study of beta testers and regular users, Communications of the ACM 61 (2) (2018) 64–71.

[8] J. Cohen, Statistical power and analysis for the behavioral sciences (2nd ed.), Lawrence Erlbaum Associates, Inc, 1988, ISBN: 0805802835.

[9] J. A. Obar, A. Oeldorf-Hirsch, The biggest lie on the internet: Ignoring the privacy policies and terms of service policies of social networking services, in: SSRN Electronic Journal, Proceedings of TPRC 44: The 44th Research Conference on Communication, Information and Internet Policy 2016, Elsevier, 2016.

[10] S. E. Taylor, Eye movements in reading: Facts and fallacies, American Educational Research Journal 2 (4) (1965) 187–202.

[11] B. J. Kwon, V. Srinivas, A. Deshpande, T. Dumitraş, Catching Worms, Trojan Horses and PUPs: Unsupervised Detection of Silent Delivery Campaigns, in: Proceedings of the 24th Network and Distributed System Security Symposium, Internet Society, 2017.

[12] P. Kotzias, S. Matic, R. Rivera, J. Caballero, Certified PUP: abuse in authenticode code signing, in: Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, ACM, 2015, pp. 465–478.

[13] P. Kotzias, L. Bilge, J. Caballero, Measuring PUP prevalence and PUP distribution through Pay-Per-Install services, in: Proceedings of the USENIX Security Symposium, 2016, pp. 739–756.

[14] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, L. F. Cranor, Crying Wolf: An Empirical Study of SSL Warning Effectiveness, in: Proceedings of the USENIX Security Symposium, 2009, pp. 399–416.

[15] E. N. Wiebe, E. F. Shaver, M. S. Wogalter, People's Beliefs about the Internet: Surveying the Positive and Negative Aspects, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 45, 2001, pp. 1186–1190.

[16] C. Bravo-Lillo, S. Komanduri, L. F. Cranor, R. W. Reeder, M. Sleeper, J. Downs, S. Schechter, Your Attention Please: Designing Security-decision UIs to Make Genuine Risks Harder to Ignore, in: Proceedings of the Ninth Symposium on Usable Privacy and Security, ACM, 2013.

[17] J. Tan, K. Nguyen, M. Theodorides, H. Negrón-Arroyo, C. Thompson, S. Egelman, D. Wagner, The Effect of Developer-specified Explanations for Permission Requests on Smartphone User Behavior, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2014.

[18] M. S. Wogalter, V. C. Conzola, T. L. Smith-Jackson, Research-based guidelines for warning design and evaluation, in: Applied ergonomics, Vol. 33, Elsevier, 2002, pp. 219–230.

[19] R. T. Gill, C. Barbera, T. Precht, A comparative evaluation of warning label designs, in: Proceedings of the Human Factors Society Annual Meeting, Vol. 31, SAGE Publications, 1987.

[20] A. P. Felt, R. W. Reeder, A. Ainslie, H. Harris, M. Walker, C. Thompson, M. E. Acer, E. Morant, S. Consolvo, Rethinking connection security indicators, in: Proceedings of the Twelfth Symposium on Usable Privacy and Security, ACM, 2016, pp. 1–14.

[21] International Organization for Standardization, ANSI Z535 2007 Revision, https://www.safetysign.com/content/maincategorycontent/ansi-z535-2007.php, accessed: 2018-02-18.

[22] M. S. Wogalter, G. A. Fontenelle, K. R. Laughery, Behavioral effectiveness of warnings, in: Proceedings of the Human Factors Society Annual Meeting, Vol. 29, SAGE Publications, 1985.

[23] C. C. Braun, L. Sansing, R. S. Kennedy, N. C. Silver, Signal word and color specifications for product warnings: an isoperformance application, in: Proceedings of the Human Factors and Ergonomics Society Annual Meeting, Vol. 38, SAGE Publications, 1994.

[24] L. T. Sharpe, A. Stockman, H. Jägle, J. Nathans, Opsin genes, cone photopigments, color vision, and color blindness, in: Color vision: From genes to perception, 1999, ISBN: 9780521004398.

[25] National Electrical Manufacturers Association, Accredited Standards Committee on Safety Signs and Colors (1998).

Figure 1: The control variant (A1) with highlighted area of PUA inquiry.

Figure 2: PUA dialog variants. The dashed line indicates omitted parts of a user dialog, such as another user dialog (LG) or Back/Install buttons.

Table 1: Attractor variants used in experiment.

Attractor type	**Variant**
A. Text structure	A1. Paragraph text (the control variant).
	A2. Bullet points.
B. Purpose stressing	B1. Providing an example.
	B2. "`What is a potentially unwanted application?`" hyperlink.
	B3. "`Why do we ask?`" hyperlink.
C. Attention raising	C1. Signal word "`Notice`" in red colour.
	C2. ANSI-inspired pictorial.
	C3. Company warning pictorial.
D. Option presentation	D1. Option order.
	D2. Option wording.
	D3. Option order and wording.
E. Combinations	E1. Combination of bullet point text structure, hyperlink and option wording.
	E2. Combination of bullet point text structure, hyperlink and option wording, without additional user dialog.
	E3. Combination of missing PUA text, hyperlink, without additional user dialog.
	E4. Combination of missing PUA text and hyperlink.

Table 2: Summary of tested variants, their PUA detection rate and statistical significance over the control variant. LG = Live Grid, another user consent dialog.

Variant	**Description**	**PUA enabl.**	**P-value**	**Signif.**	**Effect size**
A1: Control	*Paragraph text*: Text with options "Disable detection" and "Enable detection".	82.1%	–	–	–
A2	*Bulleted text*: Text description bulleted, with partial bolding.	82.6%	<.05	**Yes**	.007
B1	*Providing example*: Added an example to end of description.	81.9%	>.05	No	.005
B2	*Added hyperlink*: "What is a potentially unwanted application?"	82.2%	>.05	No	.002
B3	*Added hyperlink*: "Why do we	82.1%	>.05	No	.000

	ask?"				
C1	*Coloured signal word*: Added red *Notice*.	81.9%	>.05	No	.002
C2	*Added warning image*: ANSI-inspired pictorial.	82.1%	>.05	No	.000
C3	*Added warning image*: Company warning pictorial.	81.8%	>.05	No	.004
D1	*Option order reversed*: "Enable detection", then "Disable detection".	88.9%	<.001	**Yes**	.097
D2	*Option text changed*: from "Don't detect" to "Detect".	79.2%	<.001	**Yes**	.037
D3	*Option text changed & reversed*: Combines D1 and D2.	87.0%	<.001	**Yes**	.068

E1	*No LG & hyperlink, bulleted text, & option text changed* (combines A2, B2, D2)	80.5%	<.001	**Yes**	.020
E2	*Combines A2, B2, D2.*	80.0%	<.001	**Yes**	.026
E3	*No LG & hyperlink & no text description*: "What is a PUA?"	83.2%	<.001	**Yes**	.013
E4	*Added hyperlink & no text description*: "What is a PUA?"	81.1%	<.001	**Yes**	.015

Table 3: Summary for all tested variants, hyperlink presence, number of words and reading times, all times are in seconds.

Variant	**Contain hyperlink**	**No. of words**	**Expected reading time**	**Average time spent on screen**	**Median time spent on the screen**	**% of people who spent enough time on screen to read it**
A1: Control	No	111	26.64	15.17	8	15.1%
A2	No	107	25.68	16.27	9	17.8%
B1	No	123	29.52	15.69	8	14.2%
B2	Yes	117	28.08	15.87	9	14.1%
B3	Yes	115	27.60	16.05	9	15.3%
C1	No	112	26.88	16.03	9	16.5%
C2	No	111	26.64	15.72	9	16.0%
C3	No	111	26.64	15.65	9	15.7%
D1	No	111	26.64	13.23	8	12.0%
D2	No	108	25.92	16.03	10	16.2%
D3	No	108	25.92	14.18	9	13.5%
E1	Yes	64	15.36	21.41	14	46.3%
E2	Yes	110	25.40	18.21	11	18.3%
E3	Yes	29	6.96	20.80	11	75.9%
E4	Yes	75	18.00	19.01	9	25.3%